UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/726,779	11/29/2000	Chris Cifra	5150-43700	1986

35690 7590 01/30/2006

MEYERTONS, HOOD, KIVLIN, KOWERT & GOETZEL, P.C.
P.O. BOX 398
AUSTIN, TX 78767-0398

EXAMINER	
PILLAI, NAMITHA	

ART UNIT	PAPER NUMBER
2173	

DATE MAILED: 01/30/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/726,779	CIFRA ET AL.	
	Examiner	Art Unit	
	Namitha Pillai	2173	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *03 October 2005*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,3-14,16-27,29-40 and 42-59* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1, 3-14, 16-27, 29-40, 42-59* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Response to Amendment

1. The Examiner acknowledges Applicant's submission on 10/3/05 with amendments to claims 1, 3-14, 16-18, 20, 21, 24-27, 29-35, 37-40, 42, 44, 46-52 and 55 and the addition of new claims 56-59. All pending claims have been rejected as being previously disclosed in prior arts.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

2. Claims 1, 3-5, 7-14, 16-18, 20-27, 29-31, 33-40 and 42-59 are rejected under 35 U.S.C. 102(e) as being clearly anticipate by U. S. Patent No. 6, 298, 474 B1 (Blowers et al.).

Referring to claims 1 and 27, Blowers discloses a method for generating a computer program by receiving user input specifying a prototype, wherein the prototype comprises a series of functional operations, wherein at least one of the operations has an associated one or more parameters (column 1, lines 47-55). Blowers discloses that the input is to a prototyping application to which the user input specifies a prototype (column 3, lines 28-35). Blowers discloses functional operations representing the

functions used for carrying out tasks with a first functional operation amongst a plurality of operations with associated parameters with the functional operation (column 3, lines 15-35). Each of the multiple nodes in a structure represents functions with parameters associated with these functions. Blowers discloses automatically generating the program that implements the prototype, in response to the specified prototype, wherein comprising automatically generating a graphical user interface for the program (column 3, lines 15-20). Blowers also discloses that automatically generating the graphical user interface comprises automatically creating one or more graphical user interface elements associated with the one or more parameters of the first functional operation (column 3, lines 26-33), wherein during execution of the program, at least one of one or more graphical user interface elements are displayed and are operable to receive user input or display the output (column 9, lines 7-11), wherein as seen in Figure 7, the user configures the parameter data wherein the execution of the program is evident in the display of the graphical representations of the program along with a results "blog" window both shown in Figure 7. Blowers discloses that the program is operable to execute independently of the prototyping application (column 3, lines 35-45), there being a distinctiveness between the program that carries out the task through the structure created in the hardware configuration and the machine vision system (column 4, lines 1-15). Blowers also teaches that the user is allowed to input during execution without interference from the prototyping application thereby showing independence. Blowers teaches that users can input by single step inputs during debugging when

execution occurs along with user input, the user input independent of the prototyping application.

Referring to claims 3, 16, 29 and 42, Blowers discloses generating the program comprises automatically generating source code for the program without direct user input specifying the source code (column 2, lines 45-55).

Referring to claims 4, 17 and 30, Blowers discloses that the first functional operation has an associated input parameter (column 3, lines 28-34), wherein the graphical user interface comprises a graphical user interface element for interactively providing program input specifying a value for the input parameter, as seen in Figure 7 (column 9, lines 7-9). Blowers discloses that the automatically created graphical user interface elements include a graphical user interface element for interactively receiving user input specifying a value for the input parameter (column 8, lines 61-67), where the task sequencer interface and Blowers further teaches that the user can input or configure parameters related to these nodes of the sequence.

Referring to claims 5, 18 and 31, Blowers discloses that the first functional operation has an associated output parameter, wherein the automatically created graphical user interface elements include a graphical user interface element for viewing program output indicating a value for the output parameter (column 3, lines 45-55).

Referring to claims 7, 20 and 33, as seen in by the code formed in text forms in within the graphical representation of Figure 6 of Blowers, the generated program is text-based. Blowers also teaches that it is well known to generate programs comprising

source code for the program in a text-based programming language (column 1, lines 33-36).

Referring to claims 8, 21, 34 and 44, as seen in Figure 6, the automatically generated program is a graphical program, comprising a plurality of interconnected nodes that visually indicate functionality of the program. Blowers discloses automatically generating source code for the graphical program (column 3, lines 40-45).

Referring to claims 9, 22 and 35, Blowers discloses that the prototyping application interfaces with a programming environment application in order to perform the generation of the program (column 3, lines 40-44).

Referring to claims 10, 23 and 36, Blowers discloses determining the data type of the first parameter and automatically creating a first graphical user interface element associated with the first parameter and able to receive user input of the parameters, as seen in Figure 6, wherein the "Blob", "Acquire" are examples of controls wherein the control expresses an association based on the types of data that is being accessed.

Referring to claims 11 and 37, Blowers discloses the prototype specifying an image-processing algorithm, with the automatically generated program implementing this image-processing algorithm (column 2, lines 50-52).

Referring to claim 12, Blowers discloses automatically creating graphical user interface elements including a first graphical user interface element for receiving user input specifying a value for an input parameter of the first functional operation, wherein the value for the input parameter affects the image processing algorithm (column 3, lines 25-35).

Referring to claims 13, 26 and 39, Blowers discloses the graphical user interface including automatically created graphical user interface elements including a first graphical user interface element for viewing an output value for an output parameter of the first functional operation where the output value is determined by the image processing algorithm, as seen by the image shown in Figure 7 (column 3, lines 45-55).

Referring to claim 14, Blowers discloses a method for generating a computer program by receiving user input specifying a prototype through a prototyping environment application, wherein the prototype comprises a series of functional operations, wherein at least one of the operations has an associated one or more parameters, as seen in specifying a prototype in Figure 7 (column 2, lines 47-55 and column 9, lines 7-9). Blowers discloses the prototyping environment operable, the association caused by the specifying of the prototype by the user in the prototyping environment, wherein these parameters would be used to automatically generating the program that implements the prototype, in response to the specified prototype (column 3, lines 40-44), wherein comprising automatically generating a graphical user interface for the program (column 3, lines 15-20). Blowers also discloses that the graphical user interface comprises creating user interface controls associated with the one or more parameters (column 3, lines 1-5 and 24-30). Blowers also discloses that automatically generating the graphical user interface comprises automatically creating one or more graphical user interface elements associated with the one or more parameters (column 3, lines 26-33), wherein during execution of the program, at least one of the one or more

graphical user interface elements is displayed and is operable to receive user input or display the output (column 9, lines 7-11), wherein as seen in Figure 7, the user configures the parameter data wherein the execution of the program is evident in the display of the graphical representations of the program along with a results "blog" window both shown in Figure 7. Blowers discloses that the program is operable to execute independently of the prototyping application (column 3, lines 35-45), there being a distinctiveness between the program that carries out the task through the structure created in the hardware configuration and the machine vision system (column 4, lines 1-15). Blowers also teaches that the user is allowed to input during execution without interference from the prototyping application thereby showing independence. Blowers teaches that users can input by single step inputs during debugging when execution occurs along with user input, the user input independent of the prototyping application.

Referring to claim 24, Blowers discloses that the prototyping environment application is an image processing prototype environment application, as is seen in Figure 7 (column 11, lines 22-27). Blowers discloses the prototype specifying an image-processing algorithm, with the automatically generated program implementing this image-processing algorithm (column 2, lines 50-52).

Referring to claims 25 and 38, Blowers discloses that the automatically generated graphical user interface elements include elements that can receive input parameter values affecting the image-processing algorithm (column 4, lines 45-67).

Referring to claim 39, Blowers discloses that the automatically created graphical user interface elements include elements for viewing output parameters values determined by the image processing algorithm (column 9 lines 24-26).

Referring to claim 40, Blowers discloses a method for automatically generating a computer program by receiving a program information to a first application the program information specifying functionality of the computer program (column 1, lines 47-55). Blowers discloses automatically generating the program that implements the specified functionality, in response to the program information, wherein comprising automatically generating a graphical user interface for the program (column 3, lines 15-20). Blowers also discloses that the graphical user interface comprises creating user interface controls for providing input to and/or viewing output from the program (column 3, lines 1-5 and 24-30). Blowers also discloses that automatically generating the graphical user interface comprises automatically creating graphical user interface elements associated with the one or more parameters (column 3, lines 26-33), wherein during execution of the program, the one or more graphical user interface elements are displayed and at least one of the one or more graphical user interface elements is operable to receive user input (column 9, lines 7-11), wherein as seen in Figure 7, the user configures the parameter data wherein the execution of the program is evident in the display of the graphical representations of the program along with a results "blog" window both shown in Figure 7, wherein this user configuration of the displayed parameters would indicate receiving user input. Blowers discloses that the program is operable to execute independently of the prototyping application (column 3, lines 35-45), there being a

distinctiveness between the program that carries out the task through the structure created in the hardware configuration and the machine vision system (column 4, lines 1-15). Blowers also teaches that the user is allowed to input during execution without interference from the prototyping application thereby showing independence. Blowers teaches that users can input by single step inputs during debugging when execution occurs along with user input, the user input independent of the prototyping application.

Referring to claim 43, Blowers discloses as seen by the controls "Blob", "Acquire" on Figure 6, wherein the graphical user interface controls of the program corresponds to the parameters specified by the program information.

Referring to claim 45, Blowers discloses, as seen in Figure 6, the received program specifying a prototype, and the "if-then-else" statement specifically represent a test executive sequence and a state diagram, wherein the sequence is based on the current state of a distinct variable, as is a "if-then-else" statement is used for.

Referring to claim 46, Blowers discloses automatically generating a block diagram, wherein the block diagram comprises a plurality of interconnected nodes that visually indicate the functionality of the program as shown in Figure 6.

Referring to claim 47, Blowers discloses automatically generating a user interface panel, wherein the user interface panel comprises the graphical user interface elements (Figure 7).

Referring to claim 48, Blowers discloses receiving user input specifying a prototype, wherein the prototype comprises a series of functional operations, wherein at least one of the operations has an associated one or more parameters (column 1, lines

47-55). Blowers discloses that in response to receiving user input specifying the prototype, automatically generating a graphical program, wherein automatically generating the graphical program comprises a plurality of interconnected nodes that visually indicate functionality of the graphical program (column 8, lines 61-67). The plurality of interconnected nodes is operable to perform the series of functional operations (column 8, lines 61-67). Blowers discloses automatically generating a graphical user interface for the graphical program, wherein the graphical user interface for the graphical program comprises at leas t one graphical user interface element which is associated with at least one of the one or more parameters, wherein the graphical program is interpretable or compilable (column 9, lines 7-11).

Referring to claim 49, Blowers discloses receiving user input specifying the prototype is performed by a development environment, the graphical program comprises generating second program instructions, wherein execution of the second program instructions is independent of execution of the development environment (column 9, lines 1-25 and column 7, lines 35-39).

Referring to claim 50, Blowers discloses receiving user input specifying a prototype, wherein the prototype comprises a series of functional operations, wherein at least one of the operations has an associated one or more parameters (column 1, lines 47-55). Blowers discloses that in response to receiving user input specifying the prototype, automatically generating a graphical program, wherein automatically generating the graphical program comprises automatically generating a plurality of interconnected nodes that visually indicate functionality of the graphical program

(column 8, lines 61-67). Blowers discloses associating at least one of the one or more parameters with an element of the graphical user interface (column 9, lines 7-9).

Referring to claim 51, Blowers discloses receiving user input indicating the at least one of the one or more parameters, wherein associating at least one of the one or more parameters with the element of the graphical user interface is based on receiving user input indicating the at least one of the one or more parameters (column 3, lines 25-35).

Referring to claim 52, Blowers discloses receiving user input specifying the prototype is performed by first program instructions, the graphical program generating second program instructions, wherein execution of the second program instructions is independent of execution of the first program instructions (column 9, lines 1-25 and column 7, lines 35-39).

Referring to claim 53, Blowers discloses displaying a prototyping environment user interface on a display of a computer system, wherein the prototyping environment user interface is usable to create a prototype (Figure 7). Blowers discloses receiving user input specifying the prototype, wherein the prototype comprises a series of functional operations, wherein at least one of the operations has an associated one or more parameters (column 3, lines 25-35). Blowers discloses automatically generating a program that implements the prototype, in response to the specified prototype, automatically generating the program comprises automatically generating a graphical user interface for the program, the graphical user interface of the program comprises at least one graphical user interface element which is associated with at least one of the

associated one or more parameters, wherein the at least one graphical user interface element performs at least one of receiving information to the program and outputting information from the program during execution of the program (column 8, lines 60-67 and column 9, lines 1-25), wherein the graphical user interface of the program is independent of the prototyping environment user interface (column 7, lines 35-39).

Referring to claim 54, Blowers discloses that the program is interpretable or compilable (column 9, lines 11-12).

Referring to claim 55, Blowers discloses receiving user input to a development environment, wherein the user input specifies a series of functional operations, wherein at least one of the operations has an associated one or more parameters (column 8, lines 60-67). Blowers discloses automatically generating a program that implements the series of functional operations, in response to the user input, wherein the program execution of the program is independent of execution of the development environment (column 9, lines 1-25 and column 7, lines 35-39). Blowers discloses automatically generating the program comprises automatically generating a graphical user interface for the program, with at least one graphical user interface element which is associated with at least one of the associated one or more parameters, wherein the at least one graphical user interface element performs one or more of receiving information through the graphical user interface and outputting information from through the graphical user interface during execution of the program, wherein the program is interpretable or compilable (column 9, lines 1-25).

Referring to claim 56, Blowers discloses automatically generating the program comprises automatically generating source code to perform the functional operations of the prototype (column 4, lines 15-20). Blowers discloses that the source code is generated in a format that allows a user to edit the automatically generated source code (column 4, lines 64-67).

Referring to claim 57, Blowers discloses that functional operations also include a plurality of operations including a second functional operation with associated parameters (column 8, lines 61-67). Blowers discloses automatically generating graphical user interface elements associated with the parameters of the second functional operation (column 8, lines 61-67).

Referring to claim 58, Blowers discloses executing the automatically generated program including displaying the automatically generated graphical user interface independently of the prototyping application (column 3, lines 35-45), there being a distinctiveness between the program that carries out the task through the structure created in the hardware configuration and the machine vision system (column 4, lines 1-15).

Referring to claim 59, Blowers discloses automatically generating the program is operable to perform the functional operations of the prototype (column 8, lines 64-67).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the

invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 6, 19 and 32 are rejected under 35 U.S.C. 103(a) as being unpatentable over Blowers.

Referring to claims 6, 19 and 32, Blowers discloses that a plurality of parameters are associated with the first functional operation, wherein receiving user input specifying which of the plurality of parameters are desired to have associated graphical user interface elements (column 3, lines 28-31). Blowers discloses that the input is to a prototyping application to which the user input specifies a prototype (column 3, lines 28-35). Blowers also discloses that automatically generating graphical user interface comprises automatically creating graphical user interface elements associated with each specified parameter of the first functional operation (column 3, lines 31-35). Blowers discloses that the automatically created graphical user interface elements include a graphical user interface element for interactively receiving user input specifying a value for the input parameter (column 8, lines 61-67), where the task sequencer interface and Blowers further teaches that the user can input or configure parameters related to these nodes of the sequence. Blowers does not explicitly disclose not creating graphical user interface elements associated with unspecified parameters of the first functional operation but Blowers only discusses displaying the desired graphical user interface elements, which represent the parameters. It would have been obvious that the undesired and those not chosen by Blowers would not have graphical user interface elements associated with any parameters, for reasons that they need not be displayed. Blowers may not explicitly disclose that unnecessary graphical

user interface elements will not be generated with the unspecified parameters. But based on the fact, that an undesired parameter would not be useful to the user and need not be displayed to the user, makes it obvious that such unnecessary components would not be created. Hence, it would have been obvious to one skilled in the art, at the time of the invention to not create graphical user interface elements associated with unspecified parameters, which need not be displayed.

Response to Arguments

4. Applicant's arguments filed 10/3/05 have been fully considered but they are not persuasive.

Blowers as stated in the Applicant's arguments teaches that the task sequencer engine generated the tree structure with the desired functional tasks linked to the functionality of the structure. Blowers teaches that the user can further input parameters and input based on choosing parameters and prototypes desired. Based on the teachings of Blowers, the task sequencer engine is responsible for creating the functionality and determines the icons used in the tree structure. The cited section, column 8, lines 61-column 9, line 25 does not teach that the user has selected the graphical icons as is stated in Applicant's arguments.

The prototyping application in Blowers is the machine vision system which is separate from the components used for developing the control structure that is then used in a separate machine vision system, thereby teaching an independence between the prototyping application and the generated program. See column 8, lines 28-33.

Therefore, generation and manipulation of the tree structure representing the program is independent of the prototyping application.

Blowers teaches automatically generating source code, wherein the objective of Blowers is to alleviate a user of the system from actually writing source code, and to create an automation process for generation of the source code. See column 4, lines 15-19. The cited section also teaches how input specified by the user is used to produce this code.

Blowers discloses automatically generating a program with the task sequencer engine generating the program with the task sequence list, the list containing nodes within the tree structure of Figure 6 that are all connected. See column 8, lines 61-67. Each of the nodes within this tree structure represents functional operations that are used to carryout tasks for a machine vision system.

Blowers teaches that the user input specifying the parameters, are for parameters that are associated with desired hardware and graphical representation of machine vision tasks (column 3, lines 28-33).

Conclusion

5. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the

shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Responses to this action should be submitted as per the options cited below: The United States Patent and Trademark Office requires most patent related correspondence to be: a) faxed to the Central Fax number (571-273-8300) b) hand carried or delivered to the Customer Service Window (located at the Randolph Building, 401 Dulany Street, Alexandria, VA 22314), c) mailed to the mailing address set forth in 37 CFR 1 . 1 (e.g., P.O. Box 1450, Alexandria, VA 22313-1450), or d) transmitted to the Office using the Office's Electronic Filing System.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Namitha Pillai whose telephone number is (571) 272-4054. The examiner can normally be reached on 8:30 AM - 5:30 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Cabeca can be reached on (571) 272-4048.

All Internet e-mail communications will be made of record in the application file. PTO employees do not engage in Internet communications where there exists a possibility that sensitive information could be identified or exchanged unless the record includes a properly signed express waiver of the confidentiality requirements of 35 U.S.C. 122. This is more clearly set forth in the Interim Internet Usage Policy published

in the Official Gazette of the Patent and Trademark on February 25, 1997 at 1195 OG 89.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the Group receptionist whose telephone number is (571) 272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Namitha Pillai
Assistant Examiner
Art Unit 2173
January 20, 2006

RAYMOND J. BAYERL
PRIMARY EXAMINER
ART UNIT 2173